SECURITY
MARKING

The classified or limited status of this report applies to each page, unless otherwise marked.
Separate page printouts MUST be marked accordingly.

THE TECHNOLOGICAL INSTITUTE

NORTHWESTERN UNIVERSITY

EVANSTON, ILLINOIS

S Y S T E M S   R E S E A R C H   M E M O R A N D U M   No. 136

The Technological Institute   The College of Arts and Sciences
Northwestern University

ONE-PASS ALGORITHMS FOR SOME GENERALIZED NETWORK PROBLEMS

by

A. Charnes and W. M. Raike

August 1965

Part of the research underlying this report was undertaken for the Office of Naval Research, Contract Nonr-1228(10), Project NR 047-021, for the U. S. Public Health Service, Divisions of Water Supply and Pollution Control and Environmental Engineering and Food Protection, Contracts WP 00019-06 and EF 00355-03, and for the U. S. Army Research Office - Durham, Contract No. DA-31-124-ARO-D-322, at Northwestern University.

SYSTEMS RESEARCH GROUP
A. Charnes, Director

Introduction

The generalized network model, or problem, ([1]) and the closely related restricted dyadic problem ([1]) (called the generalized "transportation" problem in [8]) are two of the most frequently encountered special model types occurring in applications of linear programming. Although they are next in order after pure network or distribution models with respect to ease of computation, the jump in degree of difficulty is such that up to the present, at any rate, there exist no algorithms for them comparable in speed or efficiency to those for pure network or distribution problems. Yet examples abound in which some additional special structure to these generalized models facilitates solution to the extent that one expects solution methods to exist which involve little more computational effort than the pure cases. Often, too, these special structures may be part of larger or more complicated models of the same general type.

For such reasons, the development of special efficient techniques for identification and solution of any significant subclasses is an important task. We address ourselves to it from the viewpoint of the generalized network model because of the additional insight offered by the associated topological structure. Thus, in this paper we designate by topological properties two special subclasses which permit evolution of efficient algorithms. These follow by extensions of methods of Charnes and Cooper and of Dijkstra for the corresponding pure network problems. We obtain easily implemented algorithms which provide an optimum in one "pass" through the network. The proofs provided for these extended theorems differ in character from those provided (or not provided) in the more special "pure" problem algorithms published.

A (pure) network is an oriented connected graph with the following additional features: associated with each link (or arc) is not only a direction but a price, and with each node (or vertex) a quantity representing a supply or demand. Some commodity is regarded as flowing along the links from nodes at which a supply is present to nodes at which there is some demand; flow on any link produces a per unit revenue in the amount given by the price on that link. Capacitated networks, meaning networks in which there is an upper bound to the flow on each link, are not considered here. (See [1], [6].)

Such a network, having m nodes and n links, can be described by its incidence matrix A, an m x n matrix in which the j^{th} column (corresponding to link j) contains a -1 in row k, a +1 in row q, and zeros elsewhere when link j leads <u>from</u> node k <u>to</u> node q. An m-vector, b, contains in its i^{th} position the supply (with a - sign) or demand (with a + sign) associated with node i ; the n-vector, c, contains in its j^{th} position the unit price associated with link j.

If it is desired to maximize total revenue while satisfying the supply and demand restrictions, the optimal flow pattern x is the solution to the linear programming problem:

$$\text{(1)} \qquad \begin{array}{ll} \text{Maximize} & c^T x \\ \text{Subject to:} & Ax = b \\ & x \geq 0 \, , \end{array}$$

where $x = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}$, x_j being the flow on link j. There exist many variations on this theme; for example, some of the equations in (1) may be replaced by inequalities.

A <u>generalized</u> network differs from the above in that the nonzero entries in A are not required to be ± 1 , although it is still required that each column have exactly two nonzero entries which are of opposite sign. It is clear that, by appropriate scaling of the columns of A and the corresponding elements of c, an equivalent problem may be obtained in which the <u>negative</u> element in each column of A is equal to -1. The positive element in the j^{th} column of A will be denoted k_j. The flow on link j may be regarded as producing revenue in the amount $c_j x_j$ and then being subjected to amplification (or attenuation) by the factor k_j. Thus these numbers represent, in the terminology of electrical engineering, "gains" associated with each link. [1]

In its present generality, the above model includes as a special case the model dubbed "generalized transportation problem" in [7], [8] and [9], which is a special instance of the "dyadic" models discussed in [1] and [2] .

We shall restrict our attention to a more special situation; henceforth we assume that the generalized networks under discussion have the following special features:

(a) Exactly one node (the "sink") has nonzero demand, and this node will be designated as the m^{th} node. [2]

(b) Exactly one node (the "source") has nonzero supply, and this node will be designated as node 1.

(c) There is an unlimited supply at the source; this means that the first constraint in $Ax = b$ is the equational equivalent, i.e., incorporates the slack variable for,

$$\sum_{j=1}^{n} a_{1,j}\, x_j \geq -M$$

wherein M is as in the regularization techniques described in [1] an element from the Hilbert extension field; it may be thought of as representing a "sufficiently" large positive number.

Henceforth, (1) will be understood to incorporate these features.

It should be remarked that condition (c) above is by no means a weak assumption, and we should indicate some reasons for its adoption. In the pure network case, we know that the rank of the incidence matrix A is always $m-1$; in the case of a generalized incidence matrix representing a connected network, all that can be said in general is that the rank is either $m-1$ or m. (See appendix.) If A has full rank, it is in general

---

1/ See [5].

2/ This condition can be dispensed with; see the discussion of Algorithm 2.

necessary when implementing adjacent-extreme-point linear programming techniques to maintain a basis consisting of columns of A which will then contain m columns. The links represented by such columns can no longer form a so-called spanning tree for the network; any collection of m links in a network containing m nodes must necessarily contain a loop, and it is just such loops which are the source of difficulty in developing algorithms for the general problem. Assumption (c) effectively forces the first constraint to be redundant, which is a "reasonable" substitute for the convenience of having A to be of rank m-1 .

Even thus restricted, the problem still includes many matters of interest such as PERT, or critical path scheduling, models. Charnes and Cooper 1/ gave a "directed subdual method" for the pure network case of the problem, i.e., PERT networks. It should be emphasized, however, that such networks have the additional property that the relation $<$ between node pairs (i,j) determined by: " $i < j$ if there is an oriented path from node i to node j" determines a consistent strong partial ordering of the nodes; i.e., is antireflexive, antisymmetric and transitive. A pure network of the type described above is a PERT network if and only if the nodes form a lattice under the $<$ relation. The procedure given by Charnes and Cooper is valid only for such networks; if we allow ourselves to assume that this property is present, there is a direct extension of their directed subdual method to the case of a generalized PERT network, which we give here.

Algorithm 1

We solve the dual problem to (1) which is to determine an m-vector w which solves:

$$
\text{(2)} \qquad \begin{array}{ll} \text{Minimize} & w^T b \\ \text{Subject to:} & w^T A \geq c^T \\ & w_1 \leq 0 \text{ , all other } w_i \text{ unrestricted in sign,} \end{array}
$$

1/ See [3].

or as the problem appears with our assumptions,

$$\text{Minimize} \quad -w_1 M + w_m b_m$$

$$\text{(3)} \qquad \text{Subject to:} \quad \sum_{i=1}^{m} w_i a_{ij} \geq c_j \quad , \quad j=1, \ldots, n$$

$$w_1 \leq 0$$

in which $b_m > 0$ is the demand at node m, the sink.

Now duality theory assures us that if there is a finite optimum to (1) there is also a finite optimum to (3); furthermore, the optimal values of the two functionals are equal. Therefore, if (1) has a finite optimum we must have $w_1 = 0$. Henceforth for ease of reference we shall call w_j the node potential associated with node j (corresponding to the j^{th} row of A). The algorithm is as follows. First discard any node other than the source or sink which has only links leading into (or out of) that node, also discarding these links; since no flow can occur over such links, there can be no feasible solution if the network becomes disconnected by this procedure.

(i) Assign a node potential of zero to the source node.

(ii) Let $p(j) = \{\text{nodes } i: \text{ there is a link leading from } i \text{ to } j\}$, and denote by c_{ij} and k_{ij} the price and gain of each link (i,j) leading to j from some $i \in p(j)$. Select for use in (iii) any node s such that all immediate predecessors of s (i.e., all $i \in p(s)$) have been assigned node potentials.

(iii) Assign $w_s = \max\limits_{i \in p(s)} \left[\dfrac{w_i + c_{is}}{k_{is}} \right]$ and record the link (i,s) for which the maximum occurs. If the maximum is taken on for more than one (i,s) , select any one to be recorded.

(iv) If the sink has been assigned a potential, stop; otherwise, go back to (ii).

It remains to show that the algorithm is executable, that it terminates and that a unique path from source to sink is determined by the links recorded in (iii) which will yield the maximum total revenue.

Now suppose that the lattice property holds; then the nodes can be partitioned into classes Q_j where $Q_1=\{m\}, Q_2=p(Q_1),\ldots,Q_j=p(Q_{j-1}),\ldots,Q_t=p(Q_{t-1})$, in which $p(Q_j) = \{\text{nodes } i\text{: } i \text{ precedes } r \text{ for some } r \in Q_j \text{ and } i \notin Q_k \text{ for } k=1,\ldots,j\}$ and where the Q_j are disjoint. ("p(Q)" is mnemonic for "predecessor of Q".) It is clear that we may apply the algorithm by assigning potentials successively to all nodes in Q_t , then to all nodes in Q_{t-1} , etc., without any ambiguity; the procedure must then terminate after at most m steps. It is equally clear that the links recorded in (iii) contain a unique path from source to sink, since at each node exactly one link entering that node is recorded. When the sink has been assigned a potential this path is found by looking backward from the sink to its recorded predecessor, then to <u>its</u> recorded predecessor and so on until the source is reached. The flow along this path is then easily calculated; suppose the path consists of the links j_1 , j_2 ,..., j_r where $j_1 = (1,i_2)$, $j_2 = (i_2,i_3),\ldots,$ $j_r = (i_r,m)$.

Then $x_{j_r} = \dfrac{b_m}{k_{j_r}}$ and $x_{j_i} = \dfrac{x_{j_{i+1}}}{k_{j_i}}$, for $i = r,r-1,\ldots,1$. (All flows are zero for links not on the path.) It may be verified that these x_j do indeed constitute a feasible solution to the problem (1); moreover, a tedious but straightforward calculation shows that $\sum_{i=1}^{r} c_{j_i} x_{j_i} = w_m b_m$. It follows that the given flows and w_i constitute optimal solutions to (1) and (3) respectively if the node potentials w_i form a feasible solution to the dual problem (3). Upon reflection, however, it is apparent that the w_i computed by setting $w_1 = 0$, $w_j = \max\limits_{i \in p(j)} \left[\dfrac{w_i + c_{ij}}{k_{ij}}\right]$ actually do form a feasible solution to (3). For then $w_j \geq \dfrac{w_i + c_{ij}}{k_{ij}}$, or $w_j k_{ij} - w_i \geq c_{ij}$, for all $i \in p(j)$. These relations are precisely the constraints in (3) formed by the columns of

$\begin{pmatrix} c^T \\ A \end{pmatrix}$ corresponding to links (i,j) for $i \in p(j)$. Consideration of these for all j indicates that all dual constraints are indeed satisfied. The validity of Algorith 1 is thus established.

<u>Algorithm 2</u>

The more general procedure which we present here is an extension for 1-source generalized networks of a method of Dijkstra for determining the shortest path between two vertices of a graph. 1/ We develop the algorithm only for networks for which conditions (a), (b), (c) hold, but it will be clear from the proof of its validity that it is also applicable for networks which do not satify condition (a), i.e., "multi-sink" networks, since it actually provides an optimal path from the source to <u>any</u> node. The optimal flow will then be obtained by superimposing the flows which would be obtained by considering each sink separately. (Any node at which there is positive demand will be designated as a sink.) Algorithm 1 also possesses this wider scope.

Consider the <u>minimization</u> problem:

$$\begin{array}{lll} & \text{Minimize} & c^T x \\ (4) & \text{Subject to:} & A x = b \\ & & x \geq 0 \end{array}$$

which is to be understood to incorporate conditions (a), (b), (c) mentioned previously. (A maximization problem may be converted to one in which the objective is minimization by reversing the signs of the c_j .) Whereas here we <u>do</u> <u>not</u> require the quite restrictive lattice property needed for Algorithm 1 , two completely different conditions are imposed:

(d) All c_j are assumed nonnegative.

(e) For all j , $k_j \leq 1$.

Condition (e) is fairly stringent; it in effect ensures that the <u>optimal</u> path has no directed subpath leading from any node back to itself (i.e., "feedback") even though it may be possible to form such loops within the network.

1/ See [4] , in which the method is stated without proof.

In contrast, condition(d) is usually no restriction whatever. Often ad hoc methods suffice to obtain an equivalent problem in which all $c_j \geq 0$; at the very worst, all that is needed is any dual feasible solution with $w_1 = 0$ in order to implement the following specialization of a more generally applicable technique for effecting a transformation to a problem with nonnegative c_j. 1/

Suppose that some dual feasible solution $\bar{w}$ is available with $\bar{w}_1 = 0$. The modified objective function $\hat{c}^T = c^T - \bar{w}^T A$ must have all $\hat{c}_j \geq 0$ since the dual constraints $w^T A \leq c^T$ are assumed satisfied by $\bar{w}$. Moreover, the only effect of this transformation is to add a constant to the original functional $c^T x$, since

$$c^T x - \hat{c}^T x = \bar{w}^T Ax = (0, \bar{w}_2, \ldots, \bar{w}_m) \begin{pmatrix} -M \\ b_2 \\ \vdots \\ b_m \end{pmatrix} = \text{constant.}$$

In terms of the network data, this means simply that $\bar{w}_j k_{ij} - \bar{w}_i$ should be subtracted from c_{ij} to obtain the new $\hat{c}_{ij}$, all of which will then be nonnegative. (The double subscript notation has the same meaning as before.)

Prior to application of the algorithm we assume that certain nodes and links which are irrelevant to feasible solutions have been deleted. Specifically, these are any nodes other than the source and sink(s) at which there are only outbound (or only inbound) links, and also these links. If this deletion disconnects the network, or if there are only inbound links at the source or only outbound links at the sink(s), then there can be no feasible flow pattern. Similarly, links leading into the source or out of the sink may be immediately discarded. (There may be no feasible solution even if these conditions are not present; such a situation will however become evident in the course of applying the algorithm.) The procedure

---

1/ We develop this general technique elsewhere together with other significant applications.

is then as follows.

(i) Assign a node potential of zero to the source; circle the source (or otherwise denote the fact that its potential is to remain fixed hereafter).

(ii) Suppose the most recently circled node is i.
Let $s(i) = \{$nodes j: there is a link from i to j and node j is uncircled$\}$;
this is the set of immediate uncircled successors to i. If s(i) is empty, proceed immediately to (iii); otherwise, for each j in s(i), do (a) or (b) below:

(a) If j has not been previously examined,
$$\text{set } w_j = \frac{w_i + c_{ij}}{k_{ij}} \quad .$$

(b) If a value $\hat{w}_j$ has already been set for j, reset this value to
$$\min\left[\hat{w}_j \ , \left(\frac{w_i + c_{ij}}{k_{ij}}\right)\right]$$

(iii) Examine all nodes which have been assigned a potential but which are uncircled; choose one such uncircled node which has the least potential and circle it -- say this is node q. If there are no such nodes and if the sink (any sink, if there are several) is uncircled, then there is no feasible path from the source to the sink(s) and the process terminates.

Now for some immediate predecessor p of q the relation $w_q = \frac{w_p + c_{pq}}{k_{pq}}$ holds. Record the link (p,q) for which this is the case; if there is more than one, choose any such link. A unique path from the source to each circled node is now contained in the list of recorded links. If all sinks have been circled, an optimum is at hand; if some sink has not yet been circled, go back to (ii).

We must establish the validity of the algorithm; it is clear that it terminates after at most m repetitions of this process. It remains to show that it terminates either with a correct indication of infeasibility or with attainment of an optimum.

First, suppose that at some point there are no uncircled nodes which have been assigned (tentative) potentials. Also suppose that some sink is uncircled. Since at each stage in step (ii) all successors to the most recently circled node are assigned potentials, the class L of nodes which have been assigned potentials, whether circled or not, is exactly that class T of nodes for which one is guaranteed that there exists a path from the source to any member of T. In our case, L contains only circled nodes, implying that there are no uncircled successors to any node in L. But all nodes not in L are uncircled. Therefore there exists no path from any node in L to any node not in L. Since the source is always in L and we have supposed that some sink is uncircled, hence not in L, there can be no path from source to sink, and infeasibility has thereby been exhibited.

For ease of reference, we assume that one of the sinks is chosen to remain fixed throughout the following discussion; we can then speak without ambiguity of "the" sink. Now, ruling out infeasibility, suppose that the sink has been circled. Since the list of recorded links contains a path from the source to every circled node, it certainly contains a path from source to sink. Because only <u>one</u> link leading into a circled node is recorded and no link entering the source or leaving the sink is recorded, such a path can contain no loops (subpaths leading from some node back to itself). Such a path must therefore be unique. The flow along this path may be computed in the manner of Algorithm 1.

The matter now remaining is to show that the path obtained is an optimal path, i.e., whose flow pattern yields an optimum for the problem (4). This will be done indirectly; once we have shown that the final w_j constitute an optimal solution to the dual of (4), the same computation indicated for Algorithm 1 proves that the functional values for the two dual problems are equal and therefore that the obtained flow pattern is optimal for problem (4).

To show that the path obtained is dual optimal we shall show that the path to any circled node (obtained from the list of links recorded in (iii)) is optimal for a problem which consists of designating that circled node as the actual sink and hence for the problem (4).

Suppose we were to designate the source node as the sink. Since $w_1 = 0$, this is an optimal solution to the dual subproblem consisting of the source node alone. Now suppose that the collection of circled nodes at some stage contains r members, and that the circled w_j's for all such nodes represents an optimal solution to all of the dual subproblems obtained by: (1) considering all circled nodes and all links joining two circled nodes, and (2) by successively considering each circled node to be the sink. This supposition is true for $r = 1$ by the above comment. We shall show that the process of circling the $(r + 1)$-st node assigns it a w_{r+1} which is optimal for the subproblem obtained by considering all $r + 1$ circled nodes and supposing that the $(r + 1)$-st node is the sink; this induction then yields the desired result.

Let $p^*(r + 1) = \{\text{nodes } j: \ j \text{ is circled and there is a link from } j \text{ to } r+1\}$. Since we are to circle $r + 1$, this means that the circled

$$w_{r+1} = \min_{j \, \varepsilon \, p^*(r+1)} \left[\frac{w_j + c_{j,r+1}}{k_{j,r+1}} \right], \text{ or } \ k_{j,r+1} \, w_{r+1} - w_j \leq c_{j,r+1}$$

so that these dual constraints are satisfied. (Recall that in the dual to (4) the objective is maximization and that the inequalities are reversed from those in (3).) Moreover, the w_{r+1} thus chosen is the largest possible one which satisfies these restrictions. It only remains to show that the dual constraints corresponding to any links leading from $r + 1$ to a previously circled node are also satisfied, i.e., that

$k_{r+1,j} \, w_j - w_{r+1} \leq c_{r+1,j}$ for all circled j such that there is a link from $r+1$ to j. Consider any such j, and suppose node r was the node circled immediately prior to the circling of $r + 1$. If we can show that $w_j \leq w_r$ we will be finished, since the same argument will prove that $w_r - w_{r+1} \leq 0$ and hence $w_j - w_{r+1} \leq 0$ since all w_i are nonnegative by construction. Since all $c_j \geq 0$, this implies that $w_j - w_{r+1} \leq c_{r+1,j}$ so that, because $0 \leq k_{r+1,j} \leq 1$, also $k_{r+1,j} \, w_j - w_{r+1} \leq c_{r+1,j}$ as desired.

Suppose q is the q-th node circled and $q + 1$ the $(q + 1)$-st. We show inductively that $w_q \leq w_{q+1}$; this is obviously true for $q = 1$ since $w_1 = 0$.

There are two cases: either there is a link from q to q + 1 or not. In the first case we have $w_{q+1} \geq \frac{w_q + c_{q,q+1}}{k_{q,q+1}} \geq w_q$, so $w_q \leq w_{q+1}$. If there is no link from q to q + 1, consider the situation immediately before q is circled; both q and q + 1 must have been assigned potentials but have been uncircled. If $w_q > w_{q+1}$ then it would be impossible to choose node q to be circled next since the minimum of such uncircled w_i is always used to select the next node to be circled. Hence we must have $w_q \leq w_{q+1}$, which then establishes that $w_j \leq w_r$ as was needed to complete the proof above; the validity of Algorithm 2 is thus established.

Appendix

Rank of a Generalized Incidence Matrix

We consider a generalized incidence matrix C representing a connected generalized network; i.e., C has precisely 2 nonzero entries per column, and the network has the property that there is a path consisting of a sequence of links of the network (possibly with their orientation reversed) which connects any two given nodes in the network. Clearly then, for a network having m nodes and n links, C is m x n and connectedness implies $n \geq m - 1$.

<u>Theorem</u> $m \geq \text{rank}(C) \geq m - 1$.

<u>Proof</u> Rank (C) is always less than or equal to m; to show that rank $(C) \geq m - 1$ we show that the submatrix obtained by deleting any row of C has full rank, by showing that any such $m - 1$ rows are linearly independent.

Delete any row of C (this corresponds to the deletion of some node), and partition the remaining rows (nodes) into two classes, A and B, such that A contains all rows containing "singleton" elements and B contains all other rows; i.e., row i is in A if and only if it contains the only nonzero entry in some column of $\begin{pmatrix} A \\ B \end{pmatrix}$. Now suppose that there exist α_i , β_j , not all zero, such that

$$(1) \qquad \sum_{i=1}^{m_1} \alpha_i A^i + \sum_{j=1}^{m_2} \beta_j B^j = 0 \quad ,$$

where A^i (B^j) is the i-th (j-th) row of A (B) and $m_1 + m_2 = m - 1$.

The system (1) embodies n equations; by the definition of A and B there are m_1 of these (corresponding to columns, say, $k(1),\ldots,k(i),\ldots,k(m_1)$ of $\begin{pmatrix} A \\ B \end{pmatrix}$) such that $B^j_{k(i)} = 0$ for $j = 1,\ldots,m_2$ and $i = 1,\ldots,m_1$, since otherwise row B^j would be in A. Therefore, from these (unit vector) columns we obtain the m_1 equations

$$\sum_{i=1}^{m_1} \alpha_i A^i_{k(i)} = 0 \text{ , or by the singleton character of A,}$$

$\alpha_i A^i_{k(i)} = 0$ and hence $\alpha_i = 0$, for $i = 1,\ldots,m_1$.

Next, by connectedness of the network, there must be a link from <u>some</u> node in the collection represented by A to one in the collection represented by B (or vice versa), since by construction there are no links between the deleted node and the nodes of B. In other words, there must exist a column q of $\binom{A}{B}$ with one nonzero entry in one of the first m_1 positions and the other in the $(m_1 + k)$-th position, for some k. Recalling that all $\alpha_i = 0$, equations (1) become:

$$\sum_{j=1}^{m_2} \beta_j B_i^j = 0 , \quad i = 1,\dots,n \quad . \tag{2}$$

The column q just mentioned ensures that one such equation reduces to $\beta_k B_q^k = 0$ where $B_q^k \neq 0$; hence $\beta_k = 0$.

Now, switching row k from B to A and renaming β_k as α_{m_1+1} , the preceding argument employing the connectedness property of the network may be repeated until all rows have been exhausted. We obtain thereby that all $\beta_j = 0$., and hence the rows of $\binom{A}{B}$ are linearly independent.

REFERENCES

[1] Charnes, A., and W. W. Cooper, Management Models and Industrial Applications of Linear Programming, (New York: John Wiley and Sons, Inc., 1961).

[2] Charnes, A., W. W. Cooper, and M. H. Miller, "Dyadic Models and Sub-Dual Methods", Naval Research Logistics Quarterly, March 1961.

[3] Charnes, A., and W. W. Cooper, "A Network Interpretation and a Directed Subdual Algorithm for Critical Path Scheduling", The Journal of Industrial Engineering, July-August 1962, pp. 213-219.

[4] Dijkstra, E. W., "A Note on Two Problems in Connexion with Graphs", Numerische Mathematik, Band 1, Heft 5, October 1959, pp. 269-271.

[5] Jewell, W. S., "Optimal Flow through Networks with Gains", Operations Research, Volume 10 (1962), No. 4 (July-August), pp. 476-499.

[6] Ford, L. R., and D. R. Fulkerson, "Maximal Flow through a Network", Canadian Journal of Mathematics, 8, 1956, p. 399.

[7] Eisemann, K., "The Generalized Stepping Stone Method for the Machine Loading Model", Management Science, Volume 11, Number 1, September 1964, pp. 154-176.

[8] Balas, E., and P. Ivanescu, "On the Generalized Transportation Problem", Management Science, Volume 11, Number 1, September 1964, pp. 188-202.

[9] Lourie, J. R., "Topology and Computation of the Generalized Transporation Problem", Management Science, Volume 11, Number 1, September 1964, pp. 177-187.

Unclassified

Security Classification

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) Northwestern University	2a. REPORT SECURITY CLASSIFICATION Unclassified
	2b. GROUP

3. REPORT TITLE
One-Pass Algorithms for Some Generalized Network Problems

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)
Research report

5. AUTHOR(S) (Last name, first name, initial)
Charnes, Abraham and Raike, William M.

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
August 1965	16	9

8a. CONTRACT OR GRANT NO. Nonr 1228(10)	9a. ORIGINATOR'S REPORT NUMBER(S) Systems Research Memorandum No. 136
b. PROJECT NO. NR 047-021	
c.	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report) None
d.	

10. AVAILABILITY/LIMITATION NOTICES
Releasable without limitation on dissemination.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY Logistics and Mathematical Statistics Br. Office of Naval Research Washington, D. C. 20360

13. ABSTRACT

The generalized network problem and the closely related restricted dyadic problem are two special model types which occur frequently in applications of linear programming. Although they are next in order after pure network or distribution problems with respect to ease of computation, the jump in degree of difficulty is such that, in the most general problem, there exist no algorithms for them comparable in speed or efficiency to those for pure network or distribution problems. There are, however, numerous examples in which some additional special structure leads one to anticipate the existence of algorithms which compare favorably with the efficiency of those for the corresponding pure cases. Also, these more special structures may be encountered as part of larger or more complicated models.

In this paper we designate by topological properties two special structures which permit evolution of efficient algorithms. These follow by extensions of methods of Charnes and Cooper and of Dijkstra for the corresponding pure network problems. We obtain easily implemented algorithms which provide an optimum in one "pass" through the network. The proofs provided for these extended theorems differ in character from those provided (or not provided) in the more special "pure" problem algorithms published.

DD FORM 1 JAN 64 1473

Unclassified

Security Classification

Unclassified
Security Classification

14. KEY WORDS	LINK A ROLE	LINK A WT	LINK B ROLE	LINK B WT	LINK C ROLE	LINK C WT
network generalized network dyadic problem linear programming critical path algorithms PERT						

INSTRUCTIONS

1. ORIGINATING ACTIVITY: Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization *(corporate author)* issuing the report.

2a. REPORT SECURITY CLASSIFICATION: Enter the overall security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accordance with appropriate security regulations.

2b. GROUP: Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. REPORT TITLE: Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parenthesis immediately following the title.

4. DESCRIPTIVE NOTES: If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. AUTHOR(S): Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.

6. REPORT DATE: Enter the date of the report as day, month, year; or month, year. If more than one date appears on the report, use date of publication.

7a. TOTAL NUMBER OF PAGES: The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7b. NUMBER OF REFERENCES: Enter the total number of references cited in the report.

8a. CONTRACT OR GRANT NUMBER: If appropriate, enter the applicable number of the contract or grant under which the report was written.

8b, 8c, & 8d. PROJECT NUMBER: Enter the appropriate military department identification, such as project number, subproject number, system numbers, task number, etc.

9a. ORIGINATOR'S REPORT NUMBER(S): Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9b. OTHER REPORT NUMBER(S): If the report has been assigned any other report numbers *(either by the originator or by the sponsor)*, also enter this number(s).

10. AVAILABILITY/LIMITATION NOTICES: Enter any limitations on further dissemination of the report, other than those imposed by security classification, using standard statements such as:

(1) "Qualified requesters may obtain copies of this report from DDC."

(2) "Foreign announcement and dissemination of this report by DDC is not authorized."

(3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through ________________."

(4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through ________________."

(5) "All distribution of this report is controlled. Qualified DDC users shall request through ________________."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

11. SUPPLEMENTARY NOTES: Use for additional explanatory notes.

12. SPONSORING MILITARY ACTIVITY: Enter the name of the departmental project office or laboratory sponsoring *(paying for)* the research and development. Include address.

13. ABSTRACT: Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as *(TS), (S), (C), or (U)*.

There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.

14. KEY WORDS: Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, roles, and weights is optional.